



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/699,062	10/31/2003	Bryan M. Cantrill	03226/330001; SUN040156	2597
32615	7590	05/04/2006	EXAMINER	
OSHA LIANG L.L.P./SUN 1221 MCKINNEY, SUITE 2800 HOUSTON, TX 77010			MYINT, DENNIS Y	
			ART UNIT	PAPER NUMBER
			2162	

DATE MAILED: 05/04/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

**Office Action Summary**

Application No.

10/699,062

Applicant(s)

CANTRILL, BRYAN M.

Examiner

Dennis Myint

Art Unit

2162

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 31 October 2003.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-24 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-24 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 31 October 2003 is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- |   |   |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)             | 4) <input type="checkbox"/> Interview Summary (PTO-413)                     |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)    | Paper No(s)/Mail Date. _____  |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| Paper No(s)/Mail Date. _____  | 6) <input type="checkbox"/> Other: _____                                    |

### **DETAILED ACTION**

1. Claims 1-24 have been examined.

#### ***Claim Objections***

Claim 1 is objected to because in line 3, the claim recites, "storing the data in a data set an aggregation buffer", which is given the broadest, reasonable interpretation to mean "storing the data in a data set *in* an aggregation buffer".

Claim 2 is objected to because in line, the claim recites, "The method of claim 2", which should be rewritten as "The method of claim 1" instead.

Claim 11 is objected to because it recites "wherein the aggregation identifier is generating by applying a function..", which be rewritten as "wherein the aggregation identifier is generated by applying a function..".

#### ***Claim Rejections - 35 USC § 103***

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

2. Claim 1-4, 6, 10, 11, 16, 17, 18, 21, and 22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Tang et al. (U.S. Patent Application Publication Number 2003/0217130) in view of Coss et al. (U.S. Patent Application Publication Number 2003/0233380).

Referring to claim 1, Tang et al. is directed to a method for obtaining data from a kernel (Tang et al., Figure 2 and Paragraph 0037, i.e. " Kernel-level module 302 executes to compile the desired information to be communicated to user-space 301, thus avoiding the need to store network information that is not needed/desired to user-space 301."). However, Tang et al. does not explicitly disclose storing the data in a data set in an aggregation buffer using an aggregation an aggregation function.

On the other hand, Coss et al. teaches a method and system for user-defined aggregate functions in database systems without native support, wherein data is stored in a data set (Coss et al., Figure 3B "ADDR" 312 and "INTERMEDIATE RESULT" 310) in an aggregation buffer (Paragraph 0025, i.e. "keeping the intermediate results generated in each iteration of the loop in a variable stored *in memory that is external to the database.*") using an aggregation function (Coss et al., Paragraph 0023, i.e. "two new functions: ComputeAggregate and GetAggregateResults).

At the time the invention was made, it would have been obvious to a person ordinary skill in the art to combine the method Tang et al. with the feature of using a aggregate function and aggregate buffer as taught by Coss et al. so that, in the combined system the data set would be stored in an aggregation buffer using an aggregation an aggregation function. One would have been motivated to do so in order to provided "user-defined aggregation function for use with database systems that do not use recursive programming techniques" (Coss et al, Paragraph 0010).

Referring to claim 2, Tang et al. in view of Coss et al. is directed to the method of claim 1, wherein the data set comprises a key component (Coss et al., Paragraph 0028,

Art Unit: 2162

“unique identifier” and Figure 3A.i.e. “), an aggregation identifier component (Coss et al., Paragraph 0024 “identifier element” and Paragraph“ 0028, “a counter portion”. Note that Coss et al. uses both as the identifier element and sequential counter to uniquely identify the intermediate result (value) along with the pointer portion.), and a value component. (Coss et al., Paragraph 0027, i.e., “immediate result”). Particularly note what Coss et al. discloses on how said unique identifier (key component) is generated in Paragraph 0028.

Referring to claim 3, Tang et al. in view of Coss et al. is directed to the method of claim 1, wherein obtaining data comprises:

obtaining an expression (Coss et al., Paragraph 0024, “addition” and “multiplication”), a new value (Coss et al. Paragraph 0025, i.e. “intermediate results”), and an aggregation identifier (Coss et al., Paragraph 0024 “identifier element” and Paragraph“ 0028, “a counter portion”. Note that Coss et al. uses both the identifier element and sequential counter to uniquely identify the intermediate result (value) along with the pointer portion.) , and

generating a key using the expression (Coss et al., Paragraph 0024, “addition” and “multiplication”) and the aggregation identifier (Coss et al., Paragraph 0024 “identifier element” and Paragraph“ 0028, “a counter portion”). Note that the expression (such as “addition”, “multiplication”, Coss Paragraph 0024) identifies which aggregation function to use, which in turn produces a unique identifier, which is associated with intermediate result (Coss Paragraph 0027-0028).

As per claim 4, Tang et al. in view of Coss et al. is directed to the method of claim 3, wherein storing the data set comprises:

storing the key in the key component (Coss et al., Paragraph 0027, i.e., "Here, the identifier returned to the DBMS is stored in such a sequential matter, such as in a table, ....."),

storing the aggregation identifier in the aggregation identifier component (Coss et al., Paragraph 0024 "identifier element" and Paragraph 0028, "a counter portion".), and

updating a current value in the value component using the new value and the aggregation function (Coss et al., Paragraph 00227, i.e. "Each time an intermediate result is produced, it is stored...").

Referring to claim 6, Tang et al. in view of Coss et al. is directed to a method for storing data in a data set, wherein the data set comprises a key component (Coss et al., Paragraph 0028, "unique identifier" and Figure 3A.i.e. " ), an aggregation identifier component (Coss et al., Paragraph 0024 "identifier element" and Paragraph 0028, "a counter portion".), and a value component (Coss et al., Paragraph 0027, i.e., "immediate result"), comprising:

obtaining an expression, a new value, and an aggregation identifier (Coss et al., Paragraph 0024-0025 and 0027-0028. Also see the part on claim 2 above.);

generating a key using the expression and the aggregation identifier (Coss Paragraph 0027-0028); and

storing the data set in a buffer, wherein storing the data set comprises storing the key in the key component, storing the aggregation identifier in the aggregation identifier component, (Coss et al., Paragraph 0024 "identifier element" and Paragraph 0028, "a counter portion", Coss et al., Paragraph 0027, i.e., "Here, the identifier returned to the DBMS is stored in such a sequential matter, such as in a table, .....", Coss et al., Paragraph 0027, i.e., "Here, the identifier returned to the DBMS is stored in such a sequential matter, such as in a table, .....", and Coss et al. Paragraph 0027-0028) and updating a current value in the value component using the new value and an aggregation function (Coss et al., Paragraph 00227, i.e. "Each time an intermediate result is produced, it is stored...").

Referring to claim 10, Tang et al. in view of Coss et al. is directed to the method of claim 6, wherein the expression comprises an n-tuple (Coss et al., Paragraph 0024, "addition" and "multiplication"). In the method of claim 2 as taught by Tang et al. in view of Coss et al, aggregation functions such as "addition" and "multiplication" (Coss et al., Paragraph 0024) would be applied to the data from the kernel-level (Tang et al., Figure 2 and Paragraph 0037). n-tuples are inherent is said data from the kernel-level because said data could comprise data from different applications or client computers.

As per claim 11, Tang et al. in view of Coss et al. is directed to the method of claim 6, wherein the aggregation identifier is generating by applying a function to a user-defined name for an aggregation (Coss et al., Paragraph 0024 "identifier element" and Paragraph 0028, "a counter portion".) Note that Coss et al. user-defined name such as

Art Unit: 2162

“addition” and “multiplication” and then applies an arbitrary function, that is, “The identifier element (aggregation identifier)” is for example 0(zero) for addition operation, 1 (one) for multiplication and an empty set for a union operation.

Claim 16 and 17 are rejected on the same basis as claim 3.

Claim 18 is rejected on the same basis as claim 3. Note that Tang et al. teaches Tang et al. is directed to a method for obtaining data from a kernel. Therefore, the method Tang et al. in view of Coss et al. comprises an expression, which is in/from kernel.

Claim 21 is rejected on the same basis as claim 16.

Claim 22 is rejected on the same basis as claim 17.

3. Claim 5, 7-9, 12, 13, 15, 20, and 24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Tang et al. in view of Coss et al. and further in view of Larson et al. (U.S. Patent Number 6578131).

Referring to claim 5, Tang et al. in view of Coss et al. as applied to claim 4 does not explicitly disclose about using a hash table for update/search operations. However, Larson et al. teaches a method and system for scalable hash table wherein data is stored employing a hash table, comprising:

generating a hash key (Larson et al., Column 6 Line 40-55, i.e. “a hash table”.

Note that any hash table generates a hash key);

searching for a hash bucket (Larson et al., Column 6 Line 40-55, i.e. “lookup”);



Art Unit: 2162

searching for a hash chain element in the hash bucket (Larson et al., Column 7 Line 66 through Column 7 Line 28, i.e. "hash chain");

updating the value component of the data set associated with the hash chain element if a hash chain element corresponding to the key is found (Official note is taken that the concept of hash table and updating data using a hash-table search is notorious well know in the art.),

creating a new hash chain element if the hash chain element corresponding to the key is not found and updating the value associated with the new hash chain element (Official note is taken that creating new hash chain is in a hash table is notorious well know in the art.).

At the time the invention was made, it would have been obvious to a person of ordinary skill in the art to add the feature of employing a hash table to update and lookup data as taught Larson et al. to the method of Tang et al. in view of Coss et al so that, in the combined method, storing the data set further comprises:

generating a hash key using the key ( Larson et al. in view of Coss et al., i.e. Coss et al, Paragraph 0024-0025 and 0027-0028);

searching for a hash bucket corresponding to the key using the hash key (Larson et al. in view of Coss et al., i.e. Coss et al, Paragraph 0024-0025 and 0027-0028);

searching for a hash chain element in the hash bucket corresponding to the key; (Larson et al. in view of Coss et al., i.e. Coss et al, Paragraph 0024-0025 and 0027-0028)

Art Unit: 2162

updating the value component of the data set associated with the hash chain element if a hash chain element corresponding to the key is found, wherein the updating the value component comprises applying the aggregation function to the current value in the value component using the new value as input (Larson et al. in view of Coss et al. Paragraph 00227, i.e. "Each time an intermediate result is produced, it is stored...");

creating a new hash chain element if the hash chain element corresponding to the key is not found, wherein creating a new hash chain element comprises associating a new

data set with the new hash chain element, storing the key in a key component of the new data set, storing the aggregation identifier in an aggregation identifier component of the new data set, and storing an initial value in a value component of the new data set (Larson et al. in view of Coss et al, Paragraph 0024-0025 and 0027-0028 ); and

updating the value component associated with the new hash chain element, wherein the updating the value component associated with the new hash chain element comprises

applying the aggregation function to the initial value using the new value as input (Larson et al. in view of Coss et al, Paragraph 00227, i.e. "Each time an intermediate result is produced, it is stored...").

One would have been motivated to add the feature of employing hash table for data update and lookup as taught by Larson et al. because "hashing is often used to provide fast lookup of items in a cache" (Larson et al., Column 1 Line 22-28).

Claim 7 is rejected on the same basis as claim 5.

Referring to claim 8, Tang et al. in view of Coss et al. and further in view of Larson et al. is directed to the method of claim 7, wherein the hash chain element is associated with the data set using a pointer (Larson et al. Column 7 Line 66 through Column 7 Line 13, i.e. "Each entry in the hash table consists of nothing more than a pointer to a linked list containing all items hashing to that address. The linked list, or "hash chain", can either be embedded or separate. In the first case, the pointer field needed for the hash chain is embedded in the items themselves,.."). Therefore, in the combined method of Tang et al. in view of Coss et al. and further in view of Larson et al., said pointer would be pointing to the data set.

Referring to claim 9, Tang et al. in view of Coss et al. and further in view of Larson et al. is directed to the method of claim 7, wherein the new hash chain element is associated with the new data set using a pointer. As applied in claim 8 above, in the combined method of Tang et al. in view of Coss et al. and further in view of Larson et al., the new hash chain element is associated with the new data set using a pointer.

Claim 12 is rejected on the same basis as claim 7.

Claim 13 is rejected on the same basis as 5.

Claim 15 is rejected on the same basis as claim 9.

Claim 20 is rejected on the same basis as claim 15.

Claim 24 is rejected on the same basis as claim 5.

Art Unit: 2162

4. Claim 14, 19, and 23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Tang et al. in view of Coss et al. and further in view of Larson et al. and further in view of Barnett et al. (U.S. Patent Application Publication Number 2003/0159132).

Referring to claim 14, Tang et al. in view of Coss et al. and further in view of Larson as applied to claim 5 above does not explicitly recite the use of dictionaries. However, using data dictionaries along with hash tables is well known in the art. For instance, Barnett et al. teaches a method and system for conformance checking, wherein data dictionaries are used along with hash tables (Barnett et al. Paragraph 0028 and 0040).

At the time the invention was made, it would have been obvious to a person of ordinary skill in the art to add the feature of using data dictionaries along with hash tables as taught by Barnett et al. to the method and system of Tang et al. in view of Coss et al. and further in view of Larson et al. so that in resultant method and system would be directed to the method claim 12, wherein obtaining the aggregation identifier matching the value of the aggregation identifier comprises search at least one selected from the group consisting of a user-level dictionary and a kernel level dictionary. Note that Tang et al. teaches data from user-level and kernel-level.

Claim 19 is rejected on the same basis as claim 14. Note that Tang et al. teaches both user-level and kernel-level data.

Claim 23 is rejected on the same basis claim 14.

***Conclusion***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Dennis Myint whose telephone number is (571) 272-5629. The examiner can normally be reached on 8:30AM-5:30PM Monday-Friday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Breene can be reached on (571) 272-4107. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Dennis Myint

AU-2162

  
JOHN BREENE  
SUPERVISORY PATENT EXAMINER  
TECHNOLOGY CENTER 2100